



**DESARROLLO DE LOS MODULOS PROGRAMACIÓN ACTIVIDAD,
INVENTARIO PERIÓDICO, PETICIONES QUEJAS Y RECLAMOS (PQR) PARA
EL SISTEMA DE INFORMACIÓN C.O.D.E (CONTROL Y ORGANIZACIÓN
PARA LA DIRECCIÓN ESTRATÉGICA)**

FACULTAD DE INGENIERÍA

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA INFORMÁTICA
SANTIAGO DE CALI
2010**



**DESARROLLO DE LOS MODULOS PROGRAMACIÓN ACTIVIDAD,
INVENTARIO PERIÓDICO, PETICIONES QUEJAS Y RECLAMOS (PQR) PARA
EL SISTEMA DE INFORMACIÓN C.O.D.E (CONTROL Y ORGANIZACIÓN
PARA LA DIRECCIÓN ESTRATÉGICA)**

**JULIE PAULINE ROJAS PEREA
CODIGO 2045231**

**TRABAJO DE GRADO PARA OPTAR EL TITULO DE
INGENIERA INFORMÁTICA
MODALIDA PASANTIA INSTITUCIONAL**

**DIRECTOR
JAIME CHAVARRIAGA
INGENIERO DE SISTEMAS**

**UNIVERSIDAD AUTONOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERIA INFORMATICA
SANTIAGO DE CALI
2010**

Nota de aceptación

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Occidente para optar al título de **Ingeniera informática**

Presidente del Jurado

Jurado

Santiago de Cali, Marzo del 2010

AGRADECIMIENTOS

Gracias a la ejecución de este proyecto en la empresa Ingacon S.A, pude obtener la confianza para enfrentar cualquier reto de ahora en adelante y entender lo valioso e importante que es un empleado y el aporte que éste puede brindar a una empresa, afianzando los conocimientos adquiridos durante mi vida universitaria y aprendiendo otros nuevos de gran importancia en lo que actualización e integridad de un ingeniero informático, se refiere.

El autor expresa su agradecimiento a:

Ing. Marco Tulio Astudillo, Líder del área de sistemas en Ingacon S.A y asesor empresarial del proyecto, por brindarme la oportunidad de realizar mi pasantía en el área de Sistemas, por la confianza que siempre depositó en mi trabajo y la seguridad que esto me dio durante mi estadía en la empresa, de igual forma por compartirme de su conocimiento y experiencia.

Ing. Diego Fernando Valle, Capacitador en la arquitectura ASAP del proyecto en Ingacon S.A por su constante disposición, paciencia y enseñanza.

Ing. Jaime Chavarriaga, Ingeniero de sistemas, Máster en ingeniería de software, Docente Hora cátedra del programa de Ingeniería informática de la Universidad de Occidente de Cali, por su ayuda, orientación y contribución a la terminación de este trabajo de grado.

Al resto de mis compañeros de trabajo y de estudio que aún con pequeños aportes hacen que nos engrandezcamos como profesionales y como personas.

De igual forma, a mi familia por el apoyo que me brindaron durante esta etapa y por sus sabios consejos.

CONTENIDO

Pág.

RESUMEN.....	7
INTRODUCCION	1
1. TITULO	3
2. PARTICIPANTES	4
2.1 ESTUDIANTES.....	4
2.2 DIRECTOR ACADEMICO DEL PROYECTO.....	4
2.3. ASESOR EMPRESARIAL.....	4
3. PLANTEAMIENTO DEL PROBLEMA.	5
4. JUSTIFICACIÓN	8
5. ANTECEDENTES.....	10
6. MARCO TEORICO.....	12
7. OBJETIVOS	15
7.1 OBJETIVO GENERAL	15
7.2 OBJETIVOS ESPECÍFICOS.....	15
8. METODOLOGÍA	16
8.1 ETAPAS DEL PROYECTO.....	16
8.2 FASE DE INICIO	16
8.2.1 Modelado del negocio	16
8.2.2 Requisitos.....	17
8.3 FASE DE ELABORACIÓN	17
8.3.1 Análisis y Diseño	17
8.4 FASE DE CONSTRUCCIÓN.....	18
8.4.1 Implementación.....	18
8.4.2 Pruebas	18
8.4.3 Despliegue.....	19
9. DESARROLLO DEL PROYECTO	20
9.1 MODELADO DEL NEGOCIO	20
9.2 Especificación De Requerimientos De Software	20
9.2.1 Requerimientos No Funcionales	20
9.2.3 Requerimientos Funcionales	21
9.2.4 Listado de Casos de Uso.	26
9.2.5 Listado de Actores.....	30
9.2.6 Descripción o Guiones de Casos de Uso	31
9.2.6 Diagramas Relacionales de Casos de Uso	31
9.3 ANÁLISIS Y DISEÑO	31
9.3.1 Modelado del software.....	31
9.3.2 Arquitectura del Software	32

9.3.3	ASAP	32
9.3.4	Herramienta Ingravity	33
9.3.5	Modelado de la base de datos.....	37
10.	IMPLEMENTACIÓN.....	39
10.1	Base de Datos.....	39
10.2	Lenguaje utilizado en la Interfaz.....	39
10.3	Diagrama de Despliegue	39
10.3.1	Preliminares.....	40
10.4	PRUEBAS.....	41
11.	CONCLUSIONES	42
12.	RECOMENDACIONES	44
	GLOSARIO	45
	BIBLIOGRAFÍA	49
	ANEXOS	51

LISTADO DE TABLAS

Pág.

Tabla 1.	Matriz CU PQR	27
Tabla 2.	Matriz CU Programación Actividad S.O	28
Tabla 3.	Matriz CU Inventario Periódico	29

LISTADO DE FIGURAS

	Pág.
Figura 1. Proceso PQR	20
Figura 2. Arquitectura tres capas con ASAP	31
Figura 3. Acceso al Generador	32
Figura 4. Escogencia Motor Base de Datos	32
Figura 5. Paso Descripción Aplicación a Generar	33
Figura 6. Escogencia de Estilo Aplicación	33
Figura 7. Escogencia Tablas a Utilizar de la Base de Datos	34
Figura 8. Estructura directorios Aplicación	34
Figura 9. Diagrama de Despliegue	48

RESUMEN

Siendo Ingacon S.A una empresa prestadora de servicios en el sector de las telecomunicaciones, que busca en una empresa con tecnología de vanguardia y contando con un cliente tan exigente como Telefónica/Telecom, se ha trazado como un objetivo contar con herramientas que le permitan contar con información veraz y oportuna del negocio. Para ello, ha estado trabajando en la construcción de una aplicación web puede ser accedida y utilizada de forma apropiada por los empleados ubicados en las catorce (14) localidades que cuenta en el país, y que permita entregar la información de forma completa y a tiempo.

Esta aplicación, conocida como C.O.D.E (Control y Organización para la Dirección Estratégica), inició su desarrollo en 2008 y cuenta con módulos en producción que cubren una variedad de procesos al interior de la compañía. Sin embargo, muchos de los procesos de negocio y módulos del software aún están pendientes de construir. El presente proyecto pretende apoyar este proceso de desarrollo y brindar soluciones en 3 áreas: el área de control y facturación con su módulo de PQR (Peticiones, Quejas y Reclamos), el área de Logística con el módulo de Inventario Periódico y el área de Salud Ocupacional con el módulo Programación de Actividades.

INTRODUCCION

Ingacon S.A (Ingenieros Asociados Contratistas) es una compañía especializada en la prestación de servicios para la Industria de las Telecomunicaciones en Colombia y con proyección internacional. Su principal cliente es Telefónica / Telecom, a la cual se presta servicios en los departamentos de Antioquia, Caldas, Quindío, Tolima, Caquetá, Valle del Cauca, Cauca, Nariño, Putumayo, Chocó, Huila y Risaralda, teniendo sus oficinas principales en la ciudad de Cali y Bogotá.

En Octubre de 2008, la compañía inicia el proyecto C.O.D.E, Control y Organización para la Dirección Estratégica, pensando en mejorar los métodos de manejo de información y con el objetivo de ampliar su base de clientes del mismo sector y expandiendo sus negocios actuales.

Uno de los principales objetivos del proyecto CODE es la integración de todos los procesos operativos y algunos de los procesos administrativos en un solo modelo integrado de información y consolidar la totalidad de los datos requeridos para producir información y conocimiento del negocio. En particular, C.O.D.E (Control y Organización para la Dirección Estratégica) busca integrar los diferentes procesos operativos y administrativos, exceptuando los procesos contables, de tesorería y de nómina.

Técnicamente el sistema de información C.O.D.E opera sobre Web, lo que ha permitido que todas las localidades puedan acceder y trabajar sin inconvenientes desde diferentes localidades. Adicionalmente, cuenta con métodos de seguridad basados en autenticación de usuarios y manejo de perfiles de acceso y operación basada en sesiones, bajo un esquema de seguridad perimetral apoyado en un Firewall e integrado a un servicio de directorios LDAP que administra los accesos a la plataforma interna de la compañía.

A finales de 2009, el estado del proyecto C.O.D.E se ha estimado en una cobertura del 65% de los procesos. Desarrollo que incluye algunos módulos en producción: Almacén (con un 20% de avance), Datos – Instalaciones (6% de avance), Datos – Mantenimiento (6% de avance), Compras (6% de avance), Masivos (6% de avance) y el Modulo Gerencial (16% de avance), y pendientes aún los módulos de Registro de Proveedores (4%), Control y Facturación (6%), Construcción Fibra (6%) y Construcción Cobre (6%).

El presente proyecto pretende apoyar el desarrollo del proyecto C.O.D.E y aborda la construcción de un conjunto de módulos. En particular, el desarrollo de los módulos de:

- Programación de Actividades, para el área de Salud Ocupacional

- Inventario periódico, para el área de Logística
- Peticiones, Quejas y Reclamos (PQR), para el área de Control y Facturación.

Desarrollo de Software realizado empleando el lenguaje de programación PHP, la base de datos PostgreSQL, la arquitectura de software ASAP y la herramienta de generación de código Ingravity.

1. TITULO

DESARROLLO DE LOS MODULOS PROGRAMACIÓN ACTIVIDAD,
INVENTARIO PERIÓDICO, PETICIONES QUEJAS Y RECLAMOS (PQR) PARA
EL SISTEMA DE INFORMACIÓN C.O.D.E (CONTROL Y ORGANIZACIÓN PARA
LA DIRECCIÓN ESTRATÉGICA)

2. PARTICIPANTES

2.1 ESTUDIANTES

Nombres y apellidos	Código	Programa	Modalidad	e-mail
Julie Pauline Rojas Perea	2045231	Ingeniería Informática	Pasantía Empresarial	Julipa87@gmail.com

2.2 DIRECTOR ACADEMICO DEL PROYECTO

Nombres y apellidos	Títulos
Jaime Chavarriaga	<ul style="list-style-type: none">• Ingeniero de sistemas, Universidad de San Buenaventura Cali• Especialista en investigación educativa, Universidad de San Buenaventura Cali• Especialista en gerencia informática organizacional, Universidad ICESI• Máster en ingeniería de software, universidad politécnica de Madrid• Docente Hora cátedra del programa de Ingeniería informática de la Universidad Autónoma de Occidente

2.3. ASESOR EMPRESARIAL

Nombres y apellidos	Cargo
Jonier Murillo	Lider Área de Desarrollo

3. PLANTEAMIENTO DEL PROBLEMA.

Ingacon S.A. es una empresa que presta diversos servicios para el sector de las telecomunicaciones. Servicios que se prestan de forma masiva en las diferentes localidades en donde la empresa tiene sede y que requieren de la coordinación de una gran cantidad de personal. Grandes cantidades de datos son generados en la ejecución de estos procesos y se requieren de sistemas de información para su manejo eficiente.

Ingacon es proveedor de servicios para Telefónica / Telecom en la administración, ampliación y mantenimiento de la red de acceso. Servicio que requiere de la realización constante de diferentes trabajos en todo el país. Todos estos trabajos definidos por Telefónica / Telecom generan una gran cantidad de información relacionada con el personal, los clientes atendidos, la duración del trabajo y los insumos utilizados. Tradicionalmente esta información se ha manejado a través de formatos en papel, lo que dificulta su almacenamiento y análisis

La realización de estas tareas requiere del uso de una serie de insumos y materiales. Estos insumos deben ser retirados del personal encargado de la ejecución de los trabajos de diferentes almacenes que Ingacon S.A en cada una de sus sedes.

Cada almacén cuenta con su propio control de existencias. Tradicionalmente, el control de existencias de los insumos, se ha llevado de forma manual por cada almacenista. Se hace un control a través de tarjetas de kárdex, donde el almacenista toma nota de las cantidades de cada artículo. Esta información, luego es diligenciada en una serie de formatos en Microsoft Excel y son enviados al coordinador de logística.

El registro manual del control de inventario, además de ser bastante dispendioso, es propenso a errores. Adicionalmente, ha derivado en una falta de control sobre el estado de algunos elementos. Por la naturaleza misma de cada producto, algunos elementos tienen un estado. Algunos productos deben ser controlados en alguno de los posibles estados, como pueden ser: nuevos en buen estado, nuevos en mal estado, nuevos no conformes, usados en buen estado, usados en mal estado y usados no conforme. Información que en el registro manual no siempre se está actualizando y que se había tornado muy difícil de controlar.

Ingacon S.A cuenta con otros sistemas de software como SIIGO.Net, para el manejo de contabilidad y nómina, y se integra al sistema SAP de Telefónica/Telecom donde esa empresa controla la información de su operación. En estos diferentes sistemas se cuentan con códigos que identifican los diferentes productos que se manejan. Cuando un almacenista, empleado de Ingacon S.A

determina que un producto no ha sido codificado en los sistemas internos, comunica esta situación al área de sistemas. Una persona autorizada en el área de sistemas, recibe la información e ingresa la codificación de los productos tanto en el sistema de SIIGO.Net como en el sistema C.O.D.E(Control y Organización para la Dirección Estratégica).

En ocasiones, los formatos en Excel usados para el control de estos productos e inventario, no contiene diligenciada la información de estos códigos. Situación que genera un problema para el área de sistemas, ya que debe analizar la información para determinar si es un producto nuevo o no, que el nombre señalado en el formato sea el apropiado y verificar que no se genere una codificación adicional para el mismo producto. Los errores en este control, pueden derivar en reprocesos, en la recodificación manual de las órdenes de trabajo y en multas por parte de Telefónica/Telecom

Ingacon, como toda empresa colombiana, debe por ley contar con un Programa de Salud Ocupacional. Programa que tiene a cargo de la realización de actividades para la promoción de la salud y prevención de enfermedades profesionales, entre otros. Los procesos y actividades del programa, tradicionalmente han sido manejados manualmente, usando una serie de formatos en Excel.

El programa de Salud Ocupacional programa actividades como capacitaciones y jornadas de vacunación, entre otras. Sin embargo, en muchos casos los empleados de Ingacon no se dan cuenta de estas actividades, reduciendo el impacto y la eficacia de las mismas. El programa se ha propuesto la creación de una aplicación que permita realizar la programación de dichas actividades y que permita a los empleados conocer las diferentes actividades y las fechas en las cuales se van a realizar.

En el área de Control y Facturación, se realizan diferentes procesos que tienen que ver con el servicio y la facturación a los clientes. Uno de los procesos involucra el manejo de las peticiones, quejas y reclamos (conocido como PQR). Proceso que se ha llevado en forma manual, empleando correos electrónicos y formatos en Excel y papel.

El área recibe las peticiones PQR a través de correos electrónicos dirigidos al director operativo de Ingacon. Este debe revisar la información y enviar un correo electrónico a la persona encargada. Este, a su vez, envía otro correo electrónico al director del área que le corresponde la revisión y solución. Cada solicitud tiene un tiempo de respuesta definido de acuerdo a la prioridad del caso. Si la solicitud no es atendida y respondida en ese plazo, se considera que no se ha hecho una atención apropiada.

La forma manual como se han trabajado las peticiones PQR, redundando en dificultades para controlar los casos y en hacer el seguimiento apropiado para garantizar el cumplimiento de los tiempos. Si algún usuario tiene problemas con el correo electrónico o no tiene tiempo de leerlos, puede ocasionar fallas en la atención y derivar en multas por parte de Telefónica/Telecom y en la insatisfacción del cliente.

4. JUSTIFICACIÓN

Ingacon S.A, es reformada constantemente en su estructura y procesos de negocio, respondiendo a las características del contrato en donde se estipula que la compañía brindará todos los servicios a Telecom acorde a sus necesidades y cambios implementados por ellos y por las directivas de la casa matriz Telefónica.

Debido a estos constantes cambios en los procesos de negocio y la información generada por la operación, Ingacon S.A. creó el área de Sistemas el 15 de junio de 2008. Área donde uno de los principales temas es el establecimiento de un sistema de información que maneje la totalidad de los datos de la operación y se pueda integrar al sistema financiero SIIGO, de manera que los cambios propuestos por el cliente Telecom, puedan ser llevados a cabo en el menor tiempo posible, contando con datos verídicos y oportunos y logrando la reducción de las multas impuestas por TELECOM cuando se hallan errores en los informes presentados.

Para desarrollar este sistema de información, se ha definido el proyecto C.O.D.E (Control y Organización para la Dirección Estratégica). Proyecto que involucra la integración de los procesos operativos y administrativos de la compañía. Tarea que se ha estado desarrollando desde mediados de 2008 y que representa un proceso permanente de desarrollo e integración de diferentes módulos y aplicaciones.

Para satisfacer las necesidades de los clientes, se requieren de la implementación de varios módulos sistemas que en la actualidad no cuenta el proyecto.

Una solución requerida es el desarrollo de un Sistema de Inventario en el cual se manejen las variables de producto, cantidades a ordenar, costos, y estado del producto. Solución que permita, realizando una revisión o inventario periódico, identificar las adquisiciones de materiales recientes por la organización así como aquellos materiales de los cuales se necesita renovación.

Contando con un sistema de inventarios se podrán definir las reglas y procedimientos que ayudarán a la toma de decisiones respecto de fecha y cantidad que se debe solicitar de un artículo necesario para la ejecución de una tarea como la de instalación y mantenimiento de un servicio. Incluso la consecución de insumos de oficina y materiales de aseo, redundando en la minimización de los costos de la operación y en la reducción de las penalizaciones impuestas por Telecom, por no tener los artículos disponibles en el momento de atender un evento.

Por otra parte, al manejar Ingacon S.A los productos adquiridos por Telecom para dar solución a la operación y también manejar los productos adquiridos

propriadamente por Ingacon S.A, el inventario permitiría tener información respecto a los productos que hacen parte de Telecom y a los productos de Ingacon S.A.

Adicionalmente al módulo de inventarios, el manejo de peticiones, quejas y reclamos (PQR) para el área de Control y Facturación, involucra las solicitudes de los clientes en relación con todas las actividades que desarrolla Ingacon S.A. Cualquier cliente tiene la posibilidad de reportar fallas, inconsistencias del servicio, entre otros. El módulo de PQR permitirá dar solución rápida a estas peticiones, quejas, reclamos, así como también generar informes que conlleven a la identificación las áreas que más presentan PQR y cuáles son las causales que se presentan con mayor frecuencia.

Contar con información sobre la frecuencia de las PQR, ayudará a la gerencia a enfocarse en las áreas más afectadas y descubrir los motivos de dichas fallas, mejorando la calidad en el servicio al cliente y permitiendo que cada cliente ejerza su derecho de presentar peticiones, quejas y reclamos con respecto a los servicios ofrecidos por la organización.

Adicionalmente, el módulo de PQR permitirá que el área responsable de solucionar un PQR, al implementar las alertas, pueda darse cuenta que tiene a su cargo un PQR y del tiempo disponible para responder a dicho PQR, dado que cada PQR dependiendo de su prioridad tiene un tiempo máximo de respuesta.

En el área de Salud Ocupacional, el módulo de programación de actividades, contribuirá a servir como medio de comunicación entre ésta área y el empleado respecto a las actividades de promoción, programas de salud ocupacional, actividades de capacitación dirigidos a todos los niveles de la empresa, asignando actividades de acuerdo al grupo al cual pertenezca un empleado, definir las actividades de promoción y prevención que permitan mejorar las condiciones de trabajo y de salud de los empleados, permitirá el registro organizado de dichas actividades, preparar los temas a tratar en cada reunión, determinar el lugar, generar reportes de asistencia de los miembros de la actividad programada.

El desarrollo del software involucrado en este proyecto agilizará el proceso de recopilación de información, generación de informes, y a través de dichos informes, la empresa podrá tomar mejores decisiones y dar respuesta a tiempo a las exigencias que tiene para ellos Telefónica/Telecóm, así como también mejorar la productividad de los empleados al tener formatos estándares para diligenciar la información y automatizar los procesos operativos, cumpliendo con las metas de la creación de módulos que controlen la operación, reducción de cantidad de multas, información verídica y oportuna, satisfacción del cliente y del empleado, establecidas para el año 2009.

5. ANTECEDENTES

En la actualidad existen gran cantidad de programas que se encarguen de la realización de Inventarios, tales como SOFTWARE DE FACTURACION E INVENTARIO (SISTEMA DE INFORMACION COMERCIAL SIC) en Colombia que es una solución informática que lleva más de una año en desarrollo, actualmente está en su fase final, está desarrollada sobre Kylix de Borland (Only Linux) con motor de base de datos Mysql, está dirigido por ahora para el sector salud y comprende varios módulos: Contabilidad, Facturación, Tesorería, Nomina, Inventario, Compras, mantenimiento entre otros.

Encontramos también el de la empresa V & B Empresarial E.I.R.L. (Innova Soluciones Informáticas en Perú), “Hemos desarrollado un sistema de control de inventarios vía web, que consiste en almacenar la información de inventarios en un servidor web, al cual puede accederse desde cualquier computador conectado a Internet. De esta forma el personal encargado podrá registrar los ingresos y salidas de los artículos de cada almacén de la empresa. La gerencia podrá consultar en tiempo real cual es el nivel de stock de cada artículo. Los movimientos de ingreso y salida pueden definirse de acuerdo al criterio de cada empresa. El servidor web donde estará almacenada la información de inventario puede estar en nuestro sitio web o en el suyo propio (de la empresa).”[12], aún así Ingacon S.A, maneja un concepto de sus inventarios muy particular a su operación lo que hace necesario el desarrollo de una aplicación propia de la operación que Ingacon realiza.

En lo referente a PETICIONES, QUEJAS, RECLAMOS (PQR) actualmente existen entidades que a través de su portal web, permiten al usuario registrado enviar estos PQR, más no existe una aplicación que cumpla con ésta tarea, que se encargue de avisar a un usuario o área indicado sobre la existencia de este a través de mensajes.

Colciencias dedicada a la Administración de Ciencia, Tecnología e Innovación, igualmente maneja a través de su portal web la posibilidad de que un usuario registrado pueda diligenciar sus datos y las peticiones, quejas y reclamos que tenga [2].

La dirección general de sanidad militar igualmente cuenta con el método para que los usuarios realicen sus peticiones, quejas y reclamos vía web a través del portal [4]

Por último, en el campo de Salud Ocupacional, sí existen programas diseñados para este fin de programar las actividades para esta área. Pero estos harían incurrir a la organización en costos elevados de licenciamiento, como es el caso

de P.A.S.O del Consejo Colombiano de Seguridad, pero esta aplicación tiene el agravante que no es una aplicación web, lo que nos impide el poder enlazar todas las localidades con las que cuenta Ingacon. PASO El Programa Administrador de Salud Ocupacional es un software de trabajo gerencial que le permite planear, ejecutar y evaluar las actividades del programa de salud ocupacional de su empresa. Contiene los módulos de Organización empresarial, Panorama de Riesgos, Estadísticas de Accidentalidad, Indicadores de Gestión e Historia Clínica Ocupacional [3].

6. MARCO TEORICO

Con el avance en cobertura y uso de Internet a nivel mundial, las empresas han buscado aplicar esta tecnología para ser más competentes, llegar a nuevos clientes y poder trabajar remotamente en diferentes puntos del planeta. Algunas compañías han empezado a utilizar sus aplicaciones de negocio a través de la web, usando Internet o redes Intranet. Usando Internet se hace posible desplegar la aplicación en diferentes sitios sin usar redes privadas propias, con un menor costo. Muchos proveedores de sistemas corporativos como Oracle y SAP, ya proveen sistemas que son accesibles a través de web para aprovechar estas ventajas.

Para implementar aplicaciones corporativas a través de la web, las empresas puedan hacer uso de diferentes plataformas y tecnologías. Tecnologías que pueden ser propietarias o de código fuente abierto. Las tecnologías de código fuente abierto son libremente descargables desde internet y pueden ser modificadas de acuerdo a las necesidades del usuario. El uso de tecnologías de código fuente abierto, representan una alternativa técnica muy atractiva debido a la facilidad de uso y al bajo costo de implementación.

Entre las herramientas más utilizadas para construir aplicaciones que usen tecnología intranet e internet, se pueden mencionar los servidores web de Apache, lenguajes de programación como Java, PHP y Perl, y sistemas de bases de datos como PostgreSQL y MySQL, entre otros. Herramientas que cuentan con una gran base instalada, una amplia comunidad de desarrolladores que brindan soporte a través de foros y páginas web, y una gran cantidad de manuales, aplicaciones de ejemplo y aplicaciones completas disponibles en Internet.

Empleando tecnologías de código fuente abierto, es posible desarrollar las aplicaciones empresariales basadas en web que requiere la compañía. Existen herramientas para la construcción de aplicaciones, la generación de reportes, la aplicación de técnicas de auditoría, entre otros. Herramientas que pueden combinarse apropiadamente para lograr satisfacer los requerimientos de una aplicación empresarial.

Las aplicaciones empresariales basadas en web, permiten soportar los procesos de negocio de una compañía usando tecnologías web para su funcionamiento. Las aplicaciones empresariales involucran generalmente varias características especiales:

- Existe el concepto lógica de negocios, que refiere al conjunto de reglas y tareas de procesamiento definidas por el negocio para las transacciones de negocio y el procesamiento de la información.
- Administran la información del negocio, información que normalmente es bastante compleja para soportar la visión de las diferentes áreas de la compañía.
- Normalmente involucran el procesamiento de gran cantidad de datos, con consultas constantes a la base de datos y la generación frecuente de informes.
- Normalmente involucran muchos usuarios accediendo simultáneamente a la aplicación y a la información, con diferentes perfiles y diferentes permisos sobre la información.

Debido a la complejidad de este tipo de aplicaciones, varios investigadores y empresas han trabajado en encontrar técnicas y herramientas para construirlas. Entre los trabajos más relevantes, se puede mencionar existencia de un conjunto de patrones de diseño para el desarrollo de estas aplicaciones.

La arquitectura de software inicia a través de las inspiraciones de Edsger Dijkstra, David Parnas y de Fred Brooks, pero es con Dewayne Perry de AT&T Bell Laboratories de New Jersey y Alexander Wolf de la Universidad de Colorado que la arquitectura de software tuvo gran expansión. [10] Estos primeros trabajos sobre Arquitecturas de software se desarrollaron en lenguajes no orientados a objetos.

La arquitectura de software hace referencia a un conjunto de elementos de una aplicación y su interacción entre ellos, de manera que se satisfagan la mayor cantidad de requerimientos funcionales así como también los no funcionales, representando la estructura y el funcionamiento interno del sistema. Tenemos algunas de ellas como arquitecturas en capas, orientadas a objetos, orientadas a servicios, entre muchas otras más.

Hacia 1994, y bajo la influencia del trabajo sobre “lenguajes de patrones” en Arquitectura de C. Alexander, E. Gamma et al., se desarrolló un primer trabajo sobre patrones de diseño en aplicaciones orientadas a Objetos. Este trabajo recopiló varios de los mejores diseños orientados por objetos existentes hasta el momento (como el modelo MVC existente en SmallTalk) y plasmó dichos diseños en una serie de patrones que podrían ser aplicables en diferentes contextos y aplicaciones.[8]

Los primeros patrones de arquitectura para aplicaciones empresariales fueron planteados por Sun Microsystems con aplicaciones como Alur et. Al, en textos como “Core J2EE Patterns” y “Enterprise Java Programming”, publicados en 2001.

A partir de estos trabajos de Patrones de Diseño, y de desarrollos previos en materia de arquitecturas de software estructurado, algunos centros de investigación como el SEI (Software Engineering Institute) y el ISRI (Institute for Software Reasearch International) en la Universidad de Carnegie Mellon, el ISR (Institute for Software Research) en la Universidad de California y el CMIS del Centro CSIRO de Australia, entre otros, empezaron a establecer un nuevo conjunto de teorías sobre Arquitecturas de Software Orientado por Objetos y sobre el proceso de Diseño de la Arquitectura del Software desde mediados de los 90s.

Algunos desarrollos del SEI en esta materia, propuestos por P. Clements y L. Northrop entre otros, hacen referencia a la construcción de Frameworks y mejores prácticas en dominios comunes de negocios. Esta aproximación, conocida como “Prácticas de Líneas de Productos” (Product Line Practices) busca aprovechar las similitudes existentes entre los diferentes productos de software, con el fin de reducir costos de producción y mejorar la calidad de las aplicaciones resultantes.

Estos frameworks son plantillas prediseñadas que nos permiten desarrollar otro proyecto de software con una arquitectura definida, en donde podemos incluir librerías, documentos XML, logrando construir aplicaciones completas sin la necesidad de empezar desde cero.

Entre estos modelos de arquitectura y frameworks de desarrollo, puede mencionarse el framework ASAP desarrollado por el Laboratorio de Investigación para el Desarrollo de la Ingeniería de Software LIDIS de la Universidad de San Buenaventura en Cali en colaboración con empresas de ParqueSoft.[9]

Podríamos decir, que casi todos los modelos de arquitectura de software se basan en la implementación de capas que permitan ajustar los cambios de una manera más sencilla de manera que el cambio o modificación de una capa no altere a la otra y toque empezar de cero.

Tenemos entonces modelo de tres capas (Fowler / Microsoft) con sus capas de Presentación – Lógica – Acceso a Datos, Modelo de cinco capas (Brown) con las capas Presentación – Controlador/Mediador – Dominio – Mapeo de Datos – Fuentes de Datos, entre otros modelos, siendo el más popular el modelo de tres capas [6]. Pero las arquitecturas también tienen estilos y éstos vienen determinados según la manera en que se maneje la lógica del negocio, existiendo estilos orientados a objetos y el estilo procedimental aquel manejado por el framework ASAP, siendo éste último aquel que se basa en módulos transaccionales.

7. OBJETIVOS

7.1 OBJETIVO GENERAL

Desarrollar, documentar y codificar los módulos de Inventario periódico para el área de Logística, Programación de actividades para el de área Salud ocupacional, Peticiones Quejas y Reclamos (PQR) para el área de control y facturación, del Sistema Integrado de Información Corporativo C.O.D.E (Control y Organización para la Dirección Estratégica) de la compañía Ingacon S.A.

7.2 OBJETIVOS ESPECÍFICOS

1. Analizar los procesos de los departamentos o áreas implicadas.
2. Lograr control de la información de las áreas de Salud Ocupacional, Control y Facturación, Logística.
3. Levantamiento de requerimientos.
4. Análisis, diseño, documentación y desarrollo del módulo **INVENTARIO PERIODICO** para el área de Logística.
5. Análisis, diseño, documentación y desarrollo del módulo de **PROGRAMACIÓN DE ACTIVIDAD** para el área de Salud Ocupacional.
6. Análisis, diseño, documentación y desarrollo del modulo **PQR** para el área de Control y Facturación.
7. Generar un plan de pruebas que permita identificar las fallas en el proceso y la metodología a seguir para su corrección.
8. Generar el código necesario para el cumplimiento de los requerimientos del cliente en los módulos a realizar.
9. Subir los módulos al Servidor de Desarrollo donde se realizarán las pruebas que garanticen la calidad y el correcto funcionamiento del aplicativo.
10. Implementar las aplicaciones en el Servidor de Producción de la compañía a través del cual podrán acceder los usuarios o empleados de la Empresa
11. Llevar un registro de las calificaciones que los clientes dan a nuestros servicios o requerimientos a través de las peticiones, quejas y reclamos.
12. Permitir al usuario conocer las peticiones, quejas y reclamos que le han sido asignadas a su área para dar solución
13. Permitir al usuario el registro de datos según la información necesaria para cada módulo.
14. Generar informes pertinentes para cada uno de los módulos, que reflejen un análisis de los datos ingresados por el usuario y permitan a la gerencia la toma de decisiones.

8. METODOLOGÍA

Aunque al inicio del proyecto se indicó que la metodología empleada por el área de sistemas de Ingacon S.A. correspondía a la metodología RUP, durante el transcurso del desarrollo de este proyecto se encontró una situación diferente. Ingacon S.A ha definido una metodología propia muy sencilla.

8.1 ETAPAS DEL PROYECTO

En la metodología de Ingacon S.A. se plantea unas fases de inicio, elaboración y construcción:

- **En la fase de Inicio** se recoge la información necesaria para comenzar con el desarrollo, Básicamente se busca el entendimiento del comportamiento de los procesos de la empresa, conocimiento sobre la arquitectura (si no lo tiene) y realizar reuniones con las personas implicadas en el desarrollo del módulo para conocer sus requerimientos.
- **En la fase de Elaboración** se analizan los requerimientos encontrados, determinando sus prioridades y se realiza toda la parte de diagramación del diseño. En esta fase se identifican las tablas necesarias a incluir en la Base de Datos de la empresa y si requiere alguna integración con las tablas ya existentes.
- **En la fase de Construcción**, se codifica el software y se realizan las pruebas y las actas de entrega al cliente.

Estas fases y procesos son los mismos para cada módulo a desarrollarse al interior del área de sistemas de Ingacon S.A.

8.2 FASE DE INICIO

La fase de Inicio busca establecer las bases para el desarrollo de la aplicación de software al interior de Ingacon S.A.

8.2.1 Modelado del negocio

En esta fase, la tarea es la familiarización del analista-desarrollador con el funcionamiento de la empresa, logrando un conocimiento de sus procesos y una capacitación sobre la arquitectura de software utilizada para realizar las aplicaciones.

- Reunión con el jefe del área de sistemas.

- Entender la estructura y la dinámica de la organización para la cual el sistema va ser desarrollado.
- Conocimiento del negocio buscando tener un lenguaje común para todos los implicados en el proyecto.
- Identificación de lo que debe hacer exactamente el sistema.
- Entender el problema actual.
- Capacitación en ASAP. Para la realización de esta actividad, se desarrollarán actividades pequeñas consideradas como no prioritarias, para el conocimiento de la arquitectura, de la mano de un capacitador externo a la compañía

8.2.2 Requisitos

En esta línea se revisan los requisitos establecidos por el cliente de manera que estos sean claros para ambas partes.

- Reunión con la persona que requiere la aplicación.
- Análisis de las necesidades de los usuarios finales del software para determinar qué objetivos debe cubrir.
- Levantamiento de requerimientos. En este momento surge el documento de especificación de requisitos que contiene la especificación completa de lo que debe hacer el sistema, así como también las fechas de solicitud, fechas de planeación y las fechas de cumplimiento o de entrega. Se capturan las necesidades, se identifican y analizan los requisitos y se validan con el usuario hasta llegar a un acuerdo. Esto se realiza de manera iterativa cada vez que se vaya a crear un nuevo módulo. El documento indicará también los acuerdos y decisiones tomados en esta fase.

8.3 FASE DE ELABORACIÓN

En la fase de elaboración, se analizan los requerimientos y se plantea un diseño a través de casos de uso, documentos de análisis y diseño en UML(Unified Modeling Language), y se relaciona el modelo entidad relación independiente con el modelo general con el que ya cuenta la compañía.

8.3.1 Análisis y Diseño

Como parte de las tareas de análisis y diseño, se trabaja en:

- Clasificar los requerimientos según su importancia.
- Transformar los requisitos a un lenguaje técnico informático, a través de diagramas para entendimiento y desarrollo del software

- Se realizan los algoritmos necesarios para cumplir con los requerimientos del usuario.
- Adaptar el diseño al de la aplicación ya existente (C.O.D.E Coordinación y Organización para la Dirección Estratégica).
- Definición de las tablas de la Base de datos y sus relaciones.
- Entregar documento acerca de las necesidades y/o requerimientos de los interesados a través del modelo de casos de uso, diagramas UML (Unified Modeling Language o en español Lenguaje Unificado de Modelado).

8.4 FASE DE CONSTRUCCIÓN

La fase de construcción se centra en la codificación, prueba y entrega del producto de software.

8.4.1 Implementación

Como parte de estas tareas se realiza:

- Implementación del código necesario para desarrollar los módulos.
- Integración con los módulos existentes y pruebas sobre el sistema completo.

8.4.2 Pruebas

Habiendo montado los módulos en el servidor de desarrollo o de pruebas, se realizarán ensayos que permitan identificar los fallos existentes en cada módulo.

- Definición de planes de pruebas.
- Realizar pruebas haciendo uso del peor de los casos.
- Verificar que las funciones del producto de software sean según lo acordado en los requerimientos y objetivos
- Se evalúa la calidad del producto.
- Observación del cliente, donde éste interactúa con la aplicación para verificar que las funciones del producto de software sean según lo acordado en los requerimientos y objetivos y así realizar una entrega final del producto.
- Identificar los errores y defectos y asegurarse que sean informados antes de su entrega final.

8.4.3 Despliegue

Después de las pruebas en el servidor de desarrollo, se realizará su montaje en el servidor de producción en la cual tendrá acceso el usuario final. En esta etapa se realizará la entrega del producto así como también las capacitaciones necesarias.

- Realizar pruebas en el entorno final desde el cual será utilizada la aplicación
- Realizar acta de entrega al cliente.
- Proveer de manuales de usuario.
- Formar a los usuarios a través de capacitaciones sobre el uso de la aplicación y responder posibles dudas.
- Proveer ayuda y soporte a los usuarios ya sea de manera virtual a través de Skype o en forma presencial.

9. DESARROLLO DEL PROYECTO

Como parte del proyecto, se desarrollaron una serie de módulos para el sistema corporativo de Ingacon S.A, conocido como C.O.D.E (Control y Organización paa la Dirección Estratégica). En particular, los módulos de Inventario Periódico, Programación de Actividades de Salud Ocupacional y de Administración de peticiones, quejas, reclamos (PQR).

9.1 MODELADO DEL NEGOCIO

INGACON S.A, se conforma como empresa de servicios mediante alianza estratégica y unión de fortalezas de tres compañías colombianas de ingeniería especializadas en el desarrollo y ejecución de proyectos de infraestructura en telecomunicaciones, vías, saneamiento básico, electricidad y diferentes ramas de la ingeniería, bajo las normas legales y especificaciones técnicas del cliente, cumpliendo con los plazos y programas, brindando así soluciones óptimas para los clientes. Su servicio se presta en 14 localidades del país como lo son Cali, Medellín, Armenia, Popayan, Manizales, entre otros

9.2 Especificación De Requerimientos De Software

En esta sección se especificarán los requerimientos funcionales y no funcionales para el buen desarrollo de la aplicación.

9.2.1 Requerimientos No Funcionales

Los requerimientos no funcionales hacen relación a las características del sistema que aplican de manera general como un todo, más que a rasgos particulares del mismo. Estos requerimientos son adicionales a los requerimientos funcionales que debe cumplir el sistema, y corresponden a aspectos tales como la disponibilidad, mantenibilidad, flexibilidad, seguridad, facilidad de uso, etc.

- Desempeño: Los módulos diseñados e incluidos dentro del sistema de información C.O.DE. debe estar en capacidad de dar respuesta al acceso de todos los usuarios a través de internet desde cualquier lugar.
- Arquitectura: La solución debe ser 100% desarrollada haciendo uso de la arquitectura de 3 capas implementada en el framework ASAP, el lenguaje de programación PHP y Base de Datos PostgreSQL.

- **Diseño:** El diseño gráfico de los módulos deberá corresponder con el diseño con el que ya cuentan el resto de módulos, es decir con el diseño oficial para el sistema C.O.D.E de Ingacon S.A.
- **Validación de Información:** Cada módulo deberá validar la información contenida en los formularios tales como obligatoriedad de campos y si un registro es existente en la Base de Datos o no.
- **Seguridad:** Deberá controlarse el acceso a cada uno de los módulos asignando perfiles a los usuarios o roles identificados o definidos por la empresa.
- **Facilidad de Uso:** Los módulos deberán ser fáciles de utilizar y deberá brindarse entrenamiento sobre éstos.

9.2.3 Requerimientos Funcionales

Habiendo realizado reuniones con los encargados de manejar la aplicación (módulos) o los que se convertirán en administradores de la misma, se han obtenido los siguientes requerimientos funcionales los cuales están enfocados en el comportamiento interno del software. Igualmente para un mayor entendimiento de lo que se requiere para cada módulo, se explicará a continuación la descripción general de los procesos.

PROCESOS Y REQUERIMIENTOS PARA MÓDULO PQR

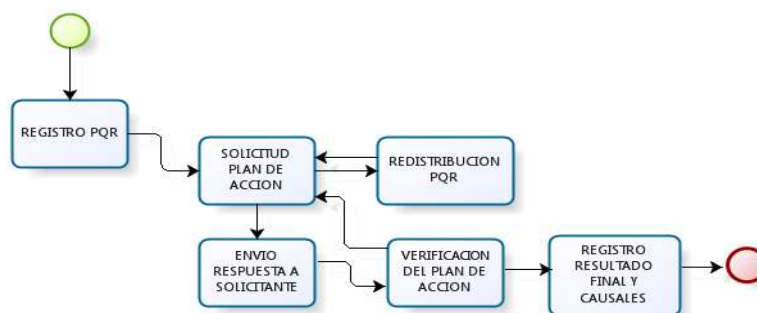


Fig. 1 Proceso PQR

Con el fin de tener un mejoramiento continuo en la prestación de servicios por parte de la organización, INGACON, cuenta con un área dedicada a PQR donde su mayor cliente Telefónica/Telecom y/o usuarios tienen la oportunidad de ejercer su derecho de presentar ante la Empresa peticiones, quejas, reclamos y penalizaciones. La creación de un módulo dispuesto para esto permitirá registrar las PQR recibidas y otorgar un responsable para ello quien determinará un plan de acción, respondiendo estos PQR en un tiempo prudente según la prioridad del caso.

El procedimiento contará con indicadores que mostrarán cómo el cliente nos percibe como organización y detectar las fallas en los servicios prestados por área.

El proyecto C.O.D.E (Control y Organización para la Dirección Estratégica). Considera el módulo PQR. Este módulo es el encargado de la administración de los PQR que llegan al área de Control y Facturación mediante correo electrónico bajo la dirección pqr@ingacon.com, también a través de medio telefónico, medio físico y/o queja en terreno.

En este momento, la persona encargada de PQR, se dispone a registrar la información perteneciente a este como lo son el tipo de solicitud, los datos del solicitante, descripción del requerimiento, entre otros. Posteriormente se procede a distribuir dicho PQR a la persona responsable para éste para que sea diligenciado el plan de acción. En caso de presentarse que el PQR sea distribuido a la persona que no corresponda, podrá realizarse una redistribución del mismo. Después de realizada la digitación del plan de acción, el PQR se responde al solicitante con dicha información. De la misma manera, si el cliente no está conforme con este plan, se procederá a otorgar uno nuevo.

Por último el encargado de diligenciar PQR, procede a registrar el resultado final y escoger las causales, procedimiento que se realiza con el fin de poder elaborar reportes respecto a las causales que no afectaron directamente a la organización.

Para el desarrollo de este módulo deben tenerse en cuenta los siguientes aspectos:

Prerequisitos:

Req No 1 Se hará uso de algunas tablas de la Base de Datos de C.O.D.E.

Req No 2 Se debe modificar la tabla persona creando una relación reflexiva que permita identificar las personas tipo naturales pertenecientes a una persona tipo jurídica.

Req No 3 Se debe modificar la interfaz existente terceros, de manera que permita ingresar personas naturales de tipo jurídico.

Req No 4 Se debe crear una nueva relación entre las tablas de SERVICIO y AREA_ORGANIZACIONAL, de manera que se pueda establecer qué área presta un servicio.

Requerimientos.

Req. No 1 El sistema debe permitir realizar registro de PQR.

Cada registro debe contener un consecutivo de PQR de la siguiente forma: PQR-ING-numero y tener asignado una orden de trabajo.

- Req. No 2** Debe poderse registrar la prioridad de PQR indicando si es alta, media, baja y asignándole un tiempo de respuesta según esta prioridad.
- Req. No 3** Debe poderse indicar el tipo de solicitud: Consulta, Petición, Queja, Reclamo, Penalización.
- Req. No 4** El Sistema debe presentarle la FECHA actual, como fecha para la construcción del PQR.
- Req. No 5** El sistema debe tener la opción para escoger un solicitante y el medio de contacto por el cual llegó el PQR.
- Req. No 6** Se debe poder registrar el detalle del requerimiento.
- Req. No 7** El sistema debe permitir asignar un área responsable.
- Req. No 8** El sistema debe contar con un filtro que según el área responsable escogida, muestre los servicios que presta esa área.
- Req. No 9** El sistema debe permitir que según el servicio se pueda escoger el sub-servicio al cuál va dirigido.
- Req. No 10** El sistema debe permitir filtrar según el área, las personas pertenecientes a ella, permitiendo escoger el nombre del responsable de dicha área.
- Req. No 11** El Sistema debe informar al responsable del PQR, mediante un correo electrónico automático que indique la existencia de un PQR a su cargo y el plazo determinado para su respuesta.
- Req. No 12** Se debe poder en caso de necesitarlo, reasignar un PQR a un área diferente.
- Req. No 13** Se debe poder registrar la descripción del plan de acción.
- Req. No 14** Se debe poder modificar la descripción del plan de acción, en caso de necesitarlo.
- Req. No 15** Se debe enviar una respuesta al solicitante, indicando cuál va a ser el plan de acción para dicho PQR.
- Req. No 16** Se debe poder registrar el resultado final de un pqr.
- Req. No 17** El sistema debe permitir registrar para un pqr sus causales.
- Req. No 18** El Sistema debe proporcionar la posibilidad de adjuntar archivos tipo PDF, GIF, JPG en la sección de registro de resultado final.
- Req. No 19** Se debe poder generar un reporte estadístico de cantidad de PQR recibidos por área.
- Req. No 20** Se debe genera un reporte de los servicios más y menos afectados por mes.
- Req. No 21** Se debe generar un reporte de cantidad de PQR recibidos por Departamento.
- Req. No 22** Se debe generar un reporte de las causales que más generan PQR.

PROCESOS Y REQUERIMIENTOS PARA MÓDULO PROGRAMACIÓN ACTIVIDADES (SALUD OCUPACIONAL)

INGACON S.A, maneja para el módulo de Salud Ocupacional, información del área para una planta de 800 trabajadores a nivel nacional y más de 20 empresas

subcontratistas con nominas de 10 hasta 50 empleados cada una, para cada una de estas localidades se generan actividades relacionadas con el concepto de salud ocupacional, de manera que este módulo se encargará de organizar dichas actividades, de manera que el usuario administrador del módulo pueda hacer ingreso de las actividades nuevas con su descripción y escogencia de un grupo al cual irá dirigido

REQUERIMIENTOS FUNCIONALES:

Req. No 1 El sistema debe permitir el ingreso de actividades relacionadas a salud ocupacional, como son capacitación, entrenamiento, reunión, entre otras.

Req. No 2 Permitir el ingreso de información sobre las actividades como costo tema, departamento y ciudad en la que es ofrecida, número de participantes, fecha de realización de la actividad, recursos humanos, hora inicio, hora fin, concepto gasto.

Req. No 3 Permitir registrar los asistentes a las actividades previamente definidas.

Req. No 4 Generar informes estadísticos sobre asistencia a las actividades por zonas, es decir por departamentos, pero también independiente (por ciudades).

Req. No 5 Permitir la realización de modificaciones del cronograma de las actividades predefinidas, tales como modificación de costos, temas, duración.

Req. No 6 Permitir consultar información del personal que haya cumplido con todas las capacitaciones.

Req. No 7 Brindarle al usuario su propia sesión con manejo restringido de acceso según el tipo de usuario.

Req. No 8 Permitir consultar el personal al que se le ha brindado capacitaciones.

Req. No 9 Permitir el ingreso de programas como lo son Copaso, medicina preventiva, medicina del trabajo, higiene industrial, seguridad industrial.

Req. No 10 El sistema debe permitir la configuración o parametrización de nuevos tipos de actividades.

Req. No 11 El sistema debe permitir clasificar las actividades según tipo capacitación, conferencia, jornada de vacunación.

Req. No 12 El sistema debe permitir la creación de grupos y asignarle personas.

Req. No 13 El sistema debe permitir que los usuarios registrados con acceso al módulo puedan ver las actividades que le corresponden.

Req. No 14 El sistema debe permitir sobre cada dato ingresado realizar operaciones de eliminación, modificación.

PROCESOS Y REQUERIMIENTOS PARA MÓDULO INVENTARIO PERIÓDICO

Para llevar a cabo las órdenes de trabajo impartidas desde Telecom a Ingacon, los almacenes de Ingacon cuentan con material propiedad telecom el cual separan en

dos conceptos como lo son telecom capex, telecom opex, siendo telecom capex los materiales diseñados para construcción y opex para instalación. Pero aparte de eso, también se cuenta con materiales dispuestos o que son propiedad de Ingacon. Por tanto para el proceso de conteo o inventario deben tenerse en cuenta tanto material Ingacon como telecom. Este módulo está diseñado para la operación de toma de inventario periódico, el cual se realiza en dos días en una semana dispuesta por el área de logística encargada de los almacenes.

Para realizar esta tarea es necesario entonces que mientras se haga el inventario, los módulos ya diseñados para el proceso de entradas y salidas (movimientos) sean desactivados para los almacenistas, dado que muchos de ellos postergan la digitación de los movimientos no haciéndolo en el tiempo correspondiente, afectando negativamente el inventario, pues posteriormente a los inventarios se hace una comparación de los saldos existentes con el inventario realizado para así proceder con los ajustes, operación que se encuentra en otro módulo no implementado en este proyecto. Estos conteos, deben realizarse dos veces con el fin de confirmar que no haya productos mal contados.

Primer Día:

1. SISTEMAS: Crea la fecha con la cual se digitará la toma de inventarios.
2. SISTEMAS: Modificará los permisos de los almacenistas, cerrando la posibilidad de digitar documentos de entrada y salida de material, de la misma forma se les dará permisos para poder digitar los datos de la toma de inventarios.
3. ALMACEN: Los almacenistas inician el primer conteo de inventarios.
4. ALMACEN: Digita el primer conteo de inventarios.

Segundo Día.

1. ALMACEN: Inicia el segundo conteo de inventarios.
2. ALMACEN: Digita el segundo conteo de inventarios.

REQUERIMIENTOS FUNCIONALES:

Req. No 1 El sistema debe permitir la realización de los conteos periódicos sobre los productos o artículos con los que cuenta cada almacén indicando la cantidad en existencia.

Req. No 2 El sistema debe permitir la creación o parametrización de los conteos.

Req. No 3 El sistema debe permitir registrar el estado de un producto como nuevo bueno, nuevo mal estado, nuevo no conforme, usado bueno, usado mal estado, usado no conforme.

Req. No 4 El sistema debe permitir mostrar los datos del usuario por sesión, como el almacén al cual pertenece y su nombre.

- Req. No 5** Se debe permitir seleccionar el número de conteo, como número uno, número dos o número tres en caso de ser necesario.
- Req. No 6** EL sistema debe permitir registrar la persona que realizó el conteo.
- Req. No 7** Se debe permitir seleccionar el producto a contar buscándolo específicamente por código o por nombre o escribir una palabra y mostrar todos los productos que la contenga.
- Req. No 8** Después de seleccionado el producto, el sistema debe mostrar la información del mismo como código SAP, código SIIGO, descripción.
- Req. No 9** El sistema debe permitir el ingreso del número de tarjeta del producto.
- Req. No 10** Se debe poder registrar el serial del producto.
- Req. No 11** El sistema debe permitir registrar el conteo número dos y cargar el conteo número uno.
- Req. No 12** El sistema debe mostrar en archivo pdf por producto el reporte de lo ingresado.
- Req. No 13** El sistema debe permitir eliminar el registro por producto ingresado.
- Req. No 14** El sistema debe permitir modificar la información ingresada de un producto, como el número de tarjeta, la cantidad según el estado y el serial.
- Req. No 15** El sistema debe permitir imprimir en formato pdf el reporte del total del conteo.
- Req. No 16** El sistema debe permitir consultar el inventario por ciudad

9.2.4 Listado de Casos de Uso.

Los casos de uso son los que me permiten identificar la interacción que tiene un usuario con la aplicación, dando también respuesta a los requerimientos exigidos por el cliente.

Dentro de los casos de uso, los más comúnmente mencionados en la mayoría de proyectos de software, son los de iniciar y cerrar sesión, dado que mi aplicación es implementada en una aplicación más grande, estos modulitos ya han sido desarrollados y por tanto no se hace necesario ingresarlos dentro de este proyecto.

Para este proyecto los casos de uso para cada aplicación son los siguientes:

PQR

- | | |
|-----------------|-----------------------|
| CU. No 1 | Ingresar PQR |
| CU. No 2 | Redistribuir PQR |
| CU. No 3 | Consultar Detalle PQR |
| CU. No 4 | Registrar Plan Acción |
| CU. No 5 | Modificar Plan Acción |

- CU. No 6** Registrar Resultado Final PQR
- CU. No 7** Registrar Causales PQR
- CU. No 8** Generar reporte de causales más frecuentes en PQR
- CU. No 9** Generar reporte de cantidad PQR por área
- CU. No 10** Generar reporte de cantidad PQR por departamento
- CU. No 11** Generar reporte de los servicios más y menos afectados por rango de fecha.

Inventario Periódico

- CU. No 1** Ingresar conteo
- CU. No 2** Consultar conteo
- CU. No 3** Modificar conteo
- CU. No 4** Eliminar conteo
- CU. No 5** Ingresar inventario
- CU. No 6** Modificar Inventario
- CU. No 7** Consultar Inventario
- CU. No 8** Generar Reporte de Inventario Digitado por artículo con seriales
- CU. No 9** Eliminar Registro por serial
- CU. No 10** Imprimir Conteo Registrado

Programación Actividad Salud Ocupacional

- CU. No 1** Registrar actividad
- CU. No 2** Registrar subprograma
- CU. No 3** Registrar tipo actividad
- CU. No 4** Registrar recurso
- CU. No 5** Registrar tipo recurso
- CU. No 6** Crear Grupo
- CU. No 7** Asignar personas a grupo
- CU. No 8** Registrar Concepto Gasto
- CU. No 9** Registrar programación actividad
- CU. No 10** Modificar actividad programada
- CU. No 11** Consultar actividades programadas según usuario
- CU. No 12** Generar reporte de Asistentes a actividad
- CU. No 13** Generar reporte de Programados a actividad y que no asistieron
- CU. No 14** Consultar Subprograma
- CU. No 15** Modificar Subprograma
- CU. No 16** Eliminar Subprograma
- CU. No 17** Consultar Tipo Recurso
- CU. No 18** Modificar Tipo Recurso
- CU. No 19** Eliminar Tipo Recurso
- CU. No 20** Consultar Recurso
- CU. No 21** Modificar Recurso
- CU. No 22** Eliminar Recurso

- CU. No 23** Consultar Tipo Actividad
- CU. No 24** Modificar Tipo Actividad
- CU. No 25** Eliminar Tipo Actividad
- CU. No 26** Consultar Actividad
- CU. No 27** Modificar Actividad
- CU. No 28** Eliminar Actividad
- CU. No 29** Consultar Concepto Gasto
- CU. No 30** Modificar Concepto Gasto
- CU. No 31** Eliminar Concepto Gasto
- CU. No 32** Consultar Grupo
- CU. No 33** Modificar Grupo
- CU. No 34** Eliminar Grupo
- CU. No 35** Eliminar Actividad Programada
- CU. No 36** Registrar Asistentes a Actividad

MATRIZ CASOS DE USO – REQUISITOS

No.	Nombre Caso de Uso	Requisitos Contemplados
CU: 01	Ingresar PQR	REQF_ 01 AL REQF_ 11
CU: 02	Redistribuir PQR	REQF_ 12
CU: 03	Consultar Detalle PQR	REQF_ 12
CU: 04	Registrar Plan Acción	REQF_ 13, REQF_ 15
CU: 05	Modificar Plan Acción	REQF_ 14, REQF_ 15
CU: 06	Registrar Resultado Final PQR	REQF_ 16, REQF_ 18
CU: 07	Registrar Causales PQR	REQF_ 17
CU: 08	Generar reporte de causales más frecuentes en PQR	REQF_ 23
CU: 09	Generar reporte de cantidad PQR por área	REQF_ 19
CU: 10	Generar reporte de cantidad PQR por departamento	REQF_ 21
CU: 11	Generar reporte de los servicios más y menos afectados por rango de fecha.	REQF_ 20

Tabla 1. Matriz CU PQR

No.	Nombre Caso de Uso	Requisitos Contemplados
CU. No 1	Registrar actividad	REQF_1
CU. No 2	Registrar subprograma	REQF_9
CU. No 3	Registrar tipo actividad	REQF_11, REQF_10
CU. No 4	Registrar recurso	REQF_2
CU. No 5	Registrar tipo recurso	REQF_2
CU. No 6	Crear Grupo	REQF_12
CU. No 7	Asignar personas a grupo	REQF_12
CU. No 8	Registrar Concepto Gasto	REQF_2
CU. No 9	Registrar programación actividad	REQF_2
CU. No 10	Modificar actividad programada	REQF_5
CU. No 11	Consultar actividades programadas según usuario	REQF_13
CU. No 12	Generar reporte de Asistentes a actividad	REQF_4, REQF_6, REQF_8
CU. No 13	Generar reporte de Programados a actividad y que no asistieron	REQF_4, REQF_8
CU. No 14	Consultar Subprograma	REQF_14
CU. No 15	Modificar Subprograma	REQF_14
CU. No 16	Eliminar Subprograma	REQF_14
CU. No 17	Consultar Tipo Recurso	REQF_14
CU. No 18	Modificar Tipo Recurso	REQF_14
CU. No 19	Eliminar Tipo Recurso	REQF_14
CU. No 20	Consultar Recurso	REQF_14
CU. No 21	Modificar Recurso	REQF_14
CU. No 22	Eliminar Recurso	REQF_14
CU. No 23	Consultar Tipo Actividad	REQF_14
CU. No 24	Modificar Tipo Actividad	REQF_14
CU. No 25	Eliminar Tipo Actividad	REQF_14
CU. No 26	Consultar Actividad	REQF_14
CU. No 27	Modificar Actividad	REQF_14
CU. No 28	Eliminar Actividad	REQF_14
CU. No 29	Consultar Concepto Gasto	REQF_14
CU. No 30	Modificar Concepto Gasto	REQF_14
CU. No 31	Eliminar Concepto Gasto	REQF_14

CU. No 32	Consultar Grupo	REQF_14
CU. No 33	Modificar Grupo	REQF_14
CU. No 34	Eliminar Grupo	REQF_14
CU. No 35	Eliminar Actividad Programada	REQF_14
CU. No 36	Registrar Asistentes a Actividad	REQF_3

Tabla 2. Matriz CU Programación Actividad S.O

No.	Nombre Caso de Uso	Requisitos Contemplados
CU. No 1	Ingresar conteo	REQF_2
CU. No 2	Consultar conteo	REQF_2
CU. No 3	Modificar conteo	REQF_2
CU. No 4	Eliminar conteo	REQF_2
CU. No 5	Ingresar inventario	REQF_1, REQF_3,REQF_4 AL REQF_11
CU. No 6	Modificar Inventario	REQF_14
CU. No 7	Consultar Inventario	REQF_16
CU. No 8	Generar Reporte de Inventario Digitado por artículo con seriales	REQF_12
CU. No 9	Eliminar Registro por serial	REQF_13
CU. No 10	Imprimir Conteo Registrado	REQF_15

Tabla 3. Matriz CU Inventario Periódico

9.2.5 Listado de Actores

Son los actores las personas que interactúan con el sistema, teniendo para esto la creación de perfiles que lo identifican y conceden o no permisos a ciertas funcionalidades del sistema. Para este proyecto tenemos los siguientes actores según aplicación:

PQR

- Administrador del Sistema
- Usuario con PQR asignado

Inventario Periódico

- Administrador del Sistema
- Almacenistas digitadores

Salud Ocupacional

- Administrador del Sistema
- Usuarios Web

9.2.6 Descripción o Guiones de Casos de Uso

Ver Anexo 3.

Anexo 3.1 Guion Caso de Uso Inventario Periódico

Anexo 3.2 Guion Caso de Uso Programación Actividad S.O

Anexo 3.3 Guion Caso de Uso PQR

9.2.6 Diagramas Relacionales de Casos de Uso

Ver Anexo 4.

Anexo 4.1 Diagrama Relacional C.U Inventario Periódico

Anexo 4.2 Diagrama Relacional C.U PQR

Anexo 4.2 Diagrama Relacional C.U Programación Actividad S.O

9.3 ANÁLISIS Y DISEÑO

9.3.1 Modelado del software

Para realizar el modelado del software dado que la metodología utilizada es una particular al negocio en la empresa, definida por el área de sistemas, se hizo uso de herramientas de tipo libre para conseguir esta tarea. En el caso de modelado de la base de datos, la herramienta Sybase Power Designer. Esta es una herramienta de modelado de datos número uno en la industria, combina varias técnicas de modelado estándar UML(Unified Modeling Language) junto con las plataformas de desarrollo más importantes como Java y Eclipse y trabaja con más de 60 sistemas de gestión de bases de datos relacionales, a través de esta herramienta se generan los modelos conceptuales, los modelos físicos para como paso final exportar dicho modelo a sql y así poderlo ejecutar en nuestra base de datos. Esta herramienta también genera reportes en donde podemos encontrar

entonces el MRD (Modelo Relacional Datos), si se han hecho comentarios sobre las tablas o los campos también podemos visualizarlos, entre otros.

9.3.2 Arquitectura del Software

La arquitectura manejada es la correspondiente a arquitectura de 3 capas. Esta es una arquitectura en la cual cada capa puede ser ejecutada en un procesador diferente, de manera tal que se puedan añadir servidores extra, convirtiéndola en una arquitectura escalable. Como su nombre lo indica cuenta con tres capas que reciben por nombre capa de presentación, capa de lógica del negocio y capa de datos. La primera es aquella donde podemos encontrar todo lo que refiere a interfaz de usuario, como formularios html, las tablas, checkbox, combobox, y es aquella que se comunica con el browser o cliente. La capa de lógica de negocio es donde establecemos las reglas de funcionamiento de una aplicación haciéndolo de manera detallada realizando validaciones, cálculos, etc, y por último la capa de datos, donde se establece la conexión a una base de datos en todo lo referente a consultas, modificaciones, entre otros, sobre ella, ejerciendo unos permisos que se han diseñado previamente sobre esta capa, de esta manera se controla el acceso a los datos, además esta capa hace a los datos independientes del servidor de aplicaciones o de la lógica del negocio.

9.3.3 ASAP

Para cada módulo debe hacerse uso del framework ASAP dado que sea la herramienta utilizada por la empresa para el desarrollo de software, por tanto cada módulo estará desarrollado bajo una arquitectura de tres capas. La arquitectura con la cual se genera las aplicaciones, será de estilo procedimental, donde el usuario a través de la interfaz (presentación) puede llamar funciones contenidas en el controlador (Lógica de Dominio) donde cada función da respuesta a un caso de uso. Cuando una de estas funciones requiera realizar operaciones sobre la Base de Datos como inserciones, eliminaciones, las solicitará a la capa de acceso a datos.

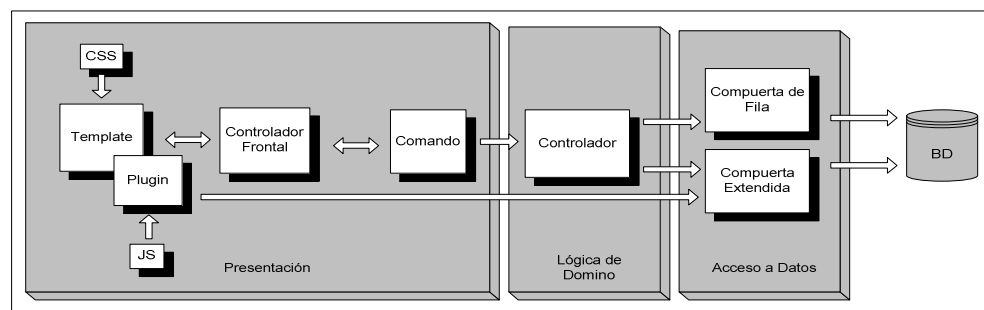


Figura 2. Arquitectura tres capas con ASAP

En esta arquitectura procedimental la lógica de dominio es organizada usando Transaction Scripts (Módulos de Transacciones), funciones que acceden a datos a través de los Data Gateways (Compuertas de Datos), como Table Gateway (Compuertas de Tabla) ó Row Gateway (Compuertas de Fila). Esta clasificación fue diseñada por Martin Flower [6]. En la capa de presentación se hace uso del Controlador Frontal, aquel que se encarga de la puesta en práctica del patrón MVC (Modelo-Vista-Control) para aplicaciones Web complejas. Este patrón es el responsable de procesar las peticiones de la aplicación. Un controlador frontal es quien aplica las plantillas, las funciones de selección y se encarga de mostrarlas a través de todas las páginas. Es quien llama al comando adecuado que ejecuta una función como añadir, eliminar, y también maneja toda la navegación de la aplicación.

En la capa de acceso a los datos se usarán Data Gateways (Compuertas de datos), aquí se definen las instrucciones SQL necesarias para obtener acceso al servidores de base de datos para su actualización.

9.3.4 Herramienta Ingravity

ASAP cuenta con una herramienta de generación llamada INGRAVITY, la cual, es el programa que genera el código para nuestra aplicación. Entonces debe accederse a ella a través del localhost en este caso y allí escoger la aplicación Generador

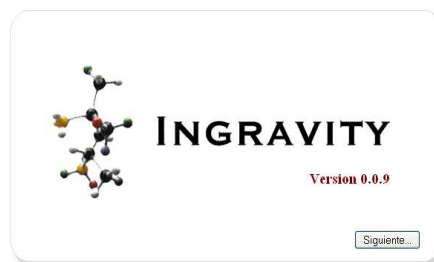


Figura. 3 Acceso al Generador

Una vez entrado aquí procedemos a escoger el motor de base de datos a utilizar e ingresaremos información del server, usuario y contraseña de la base de datos.

Capa ADOdb

Tipo de Motor: ☒ PostgreSQL ☐ Mysql

Drivers Nativos de PHP

Server: localhost

Username: postgres

Password: ●●●●●●

Resultado

Siguiete Cancelar

Figura. 4 Escogencia Motor Base de Datos

* Nombre Catalogo : INGACON

* Nombre Empresa : INGACON

* Nombre Aplicacion : ALMACEN

Nombre de la Base de Datos

Prefijo de cada archivo de la aplicación

Nombre de la aplicación. Este irá seguido del prefijo de cada archivo.

Resultado

Siguiete Cancelar

Figura. 5 Paso descripción aplicación a generar



Figura. 6 Escogencia de estilo aplicación

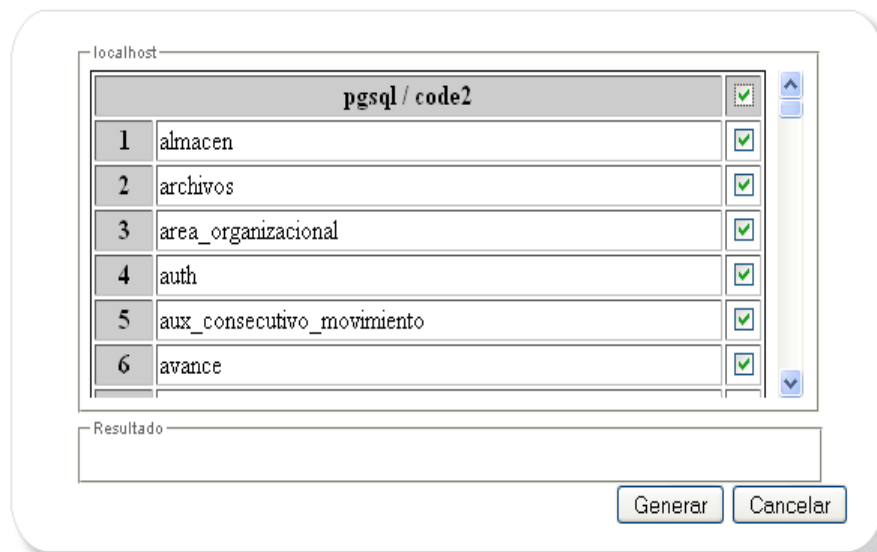


Figura. 7 Escogencia tablas a utilizar de la Base de datos

ASAP, después de generar la aplicación, creará la siguiente estructura de directorios.

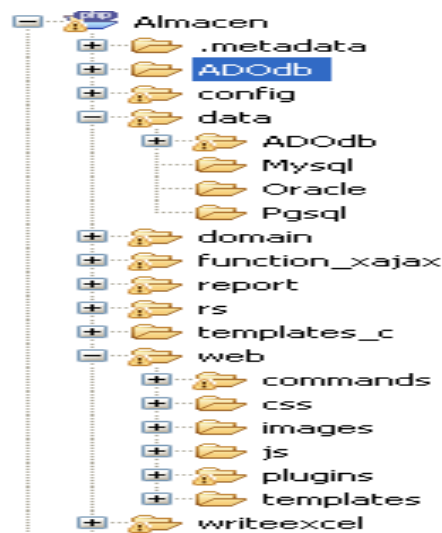


Figura. 8 Estructura directorios Aplicación

Cada carpeta contiene archivos que cumplen con funciones específicas. Tenemos la carpeta config donde encontramos como su nombre lo indica los archivos de configuración de la aplicación, tales como savenavigation y el saveconfiguration. El saveconfiguration indica qué motor de bases de datos se ha de utilizar, así como el nombre de la base de datos, usuario y la contraseña y el servidor al cual se va a conectar. El savenavigation indica los comandos y formularios existentes y a qué formulario debe direccionar un comando.

En la carpeta DATA, están contenidos los componentes de acceso a datos de la aplicación. Tenemos el directorio ADODB, que son las diferentes funciones donde encontramos el código SQL de inserción, eliminación, modificación entre otros métodos. Igualmente incluye un directorio que da soporte según la base de datos que utilicemos, estos son Mysql Oracle, Pgsql.

En el Domain encontramos los archivos manager que podemos identificar como la lógica de dominio.

Carpeta web, que a su vez contiene la carpeta commands. Cada comando es una función realizada a una tabla de la base de datos, ejemplo:

Tenemos la tabla cargos, entonces se genera un comando AddCargos que incluirá cada uno de los parámetros de esta tabla, igualmente generará comando como delete, clear, update, showlist y otros directorios que podrán ser generados por el programador.

Carpeta css contiene los cascading style sheets, Images contiene los archivos tipo imagen, JS contiene los java scripts, Plugins contiene los plugins o aquellos pequeños programas que proporcionan alguna funcionalidad específica más compleja y por último templates que son los formularios o la interfaz gráfica, realizados en código html, que finalmente verá el usuario,

De igual forma el programador podrá crear nuevos directorios, tal como es el caso de la imagen anterior donde se crea el directorio report, donde están contenidos los reportes diseñados según los requerimientos del cliente.

9.3.5 Modelado de la base de datos.

Para este proyecto se utiliza una base de datos relacional pues ésta se basa en una percepción de un mundo real que consiste en un conjunto de elementos básicos llamados entidades y relaciones entre estos elementos, brindando ventajas como que se haga más sencillo modificar o realizar cualquier operación sobre la base de datos.

Se presentarán a continuación entonces los siguientes diagramas como son el MRD (Modelo Relacional de Datos) y MER (Modelo Entidad relación).

DEFINICIÓN DE LOS MRD Y MER.

MER: son una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades. [11]

Con este modelo podremos identificar de una manera sencilla cada entidad o tabla con la que cuenta nuestra base de datos y de igual forma las relaciones y/o dependencias entre ellas.

Ver anexo 1.

- Anexo 1.1 MER Inventario Periódico
- Anexo 1.2 MER PQR
- Anexo 1.3. MER S.O

MRD:

Con este modelo se identificarán los atributos de las entidades, indicándose sus columnas, tipos de datos, si son llave foránea, llave única, etc. En conclusión mostrará el comportamiento dentro de la Base de Datos.

Ver anexo 2.

Anexo 2.1. MRD Inventario Periódico

Anexo 2.2. MRD PQR

Anexo 2.3. MRD Programación ACT S.O

10. IMPLEMENTACIÓN

10.1 Base de Datos

El área de sistemas de la empresa para manejar sus datos, recurrió a la Base de datos PostgreSQL como RDBMS (Sistema Administrador de Base de datos Relacional). Siendo INGACON una mediana empresa, con un flujo de información moderado, es postgres una buena alternativa. Por otra parte postgresQL es de tipo libre lo que redundo en que la empresa no tendría que incurrir en altos costos como si lo tendría que hacer si hubiese escogido una base de datos como Oracle. Además PostgreSQL permite store procedures, herramientas de migración de manera que en el momento que sea necesario la empresa podrá migrar a una base de datos de mayor envergadura. Cuenta con herramientas gráficas como PgAdmin, que hacen su administración más sencilla, igualmente en términos de hardware PostgreSQL no es muy exigente.

10.2 Lenguaje utilizado en la Interfaz.

Teniendo en cuenta que para nuestro desarrollo se utilizará el Framework ASAP, tal como he mencionado anteriormente, éste genera unos Templates o formularios por cada tabla de nuestra base de datos (aunque igualmente podemos diseñar nuevos formularios). Estos Templates son netamente diseñados en código html, también permiten el llamado a plugines y a CSS. La empresa cuenta ya con un estilo de CSS diseñado, el cual es el que deberá usarse para el diseño de nuestra aplicación. Dicho CSS con estilo sencillo, muestra el logo de la empresa y el árbol de menú. Por último los frames se manejan a través de un plugin el cual mezcla código PHP con código HTML.

10.3 Diagrama de Despliegue

Un diagrama de despliegue tal como su nombre lo indica muestra dónde se va a desplegar el sistema para su ejecución en un ambiente real, representando los componentes físicos como nodos estos muestran un recurso computacional que por lo general posee memoria y capacidad de procesamiento, puede ser un procesador o un dispositivo sobre el que se pueden desplegar los componentes.

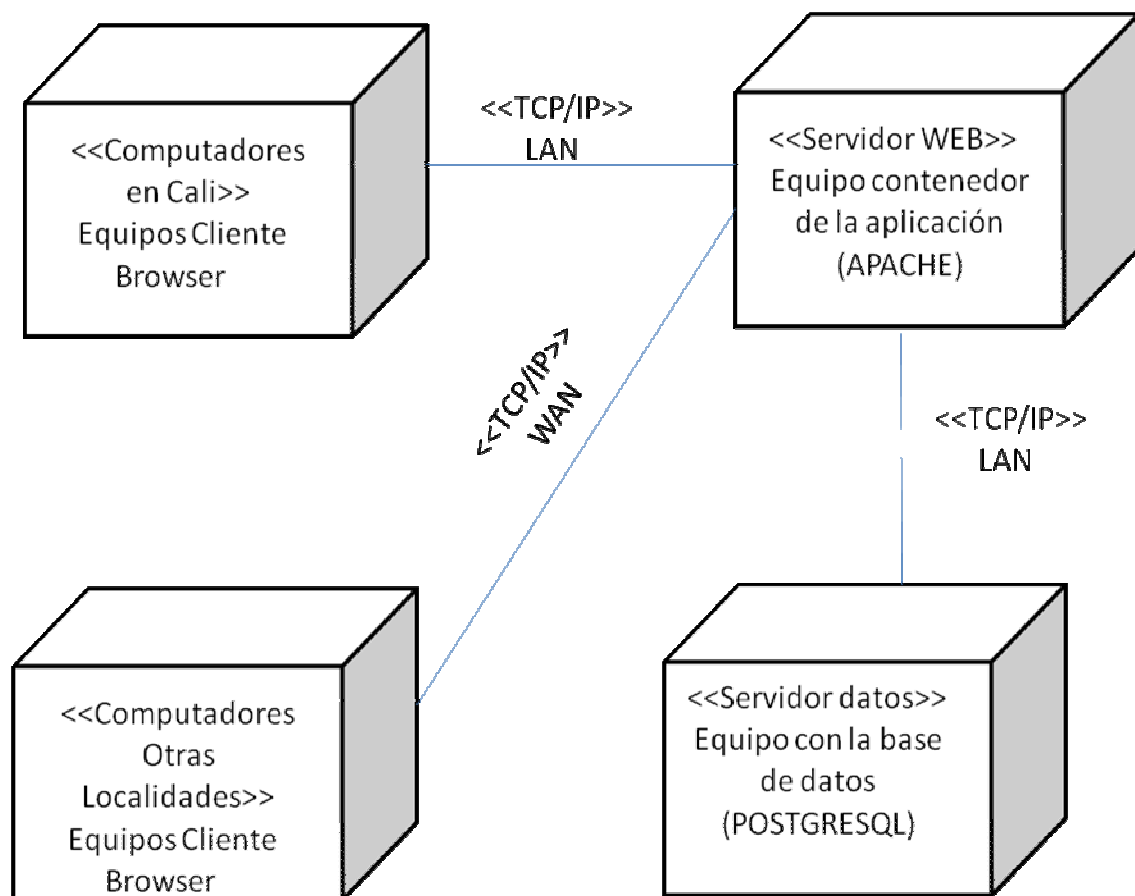


Figura. 9 Diagrama de Despliegue

10.3.1 Preliminares

Ver anexo 5.

Anexo 5.1 Preliminares Inventario Periódico

Anexo 5.2 Preliminares PQR

Anexo 5.3 Preliminares Programación Actividad Salud Ocupacional

10.4 PRUEBAS

Las pruebas dentro de un desarrollo de software son un elemento importante para determinar el buen funcionamiento de la aplicación pero ante todo el cumplimiento estricto de los requerimientos del cliente. Las pruebas constan principalmente de la interacción del usuario con el sistema, implementando para ello valores de prueba y así determinar si los resultados arrojados son los que se esperan, valores que en general hacen determinar si la prueba tuvo éxito o no, considerando una prueba exitosa aquella que me muestra fallas, pues así podrán ser corregidas antes de hacer la entrega final de la aplicación al cliente.

En el caso de INGACON las pruebas se realizan no teniendo un plan como tal previamente establecido sino contando con el servidor de desarrollo. En él se montan las aplicaciones y se hacen pruebas de funcionamiento, es decir, verificando que operaciones como el guardado, consultas, eliminaciones e inserciones se realicen correctamente. Una vez hecho esta prueba, se montan las aplicaciones en el servidor de producción, en donde se muestra entonces al usuario que va a hacer uso de dicha aplicación su funcionamiento, pudiéndole llamar a esta prueba como prueba de aceptación en la que pueden surgir correcciones o incluso nuevos requerimientos. Estos nuevos requerimientos sólo se llevan a cabo una vez establecida una conversación con el director del proyecto C.O.D.E(Control y Organización para la Dirección Estratégica). Una vez aceptada se genera un plan piloto, lo que indica que se habilita el módulo sólo en ciertas zonas para su uso y estas zonas comunican el buen funcionamiento de la aplicación e internamente sistemas revisa en la base de datos la inserción o no de información. Terminada la prueba piloto la cual dura aproximadamente dos semanas, se procede a liberar los módulos para todas las localidades, recuérdese que INGACON S.A está presente en 14 localidades del país como Medellín, Neiva, Dosquebradas, etc.

Al liberarse se procede entonces a la capacitación de los usuarios de manera presencial cuando es en Cali y de manera virtual vía skype para el resto de localidades o montando en la intranet los videos de capacitación de los módulos.

11.CONCLUSIONES

El desarrollo del proyecto resultó en una experiencia muy enriquecedora a nivel conceptual y a nivel personal. Como parte del proyecto, la autora aprendió un nuevo lenguaje de desarrollo y el uso de un framework de desarrollo, participó en varios procesos de capacitación, desarrollo de software e interacción con usuarios y desarrolladores. Adicionalmente, se tuvo la experiencia de desplegar la aplicación, participar en el proceso de montaje y de constatar el uso de las aplicaciones por parte de los usuarios.

A partir del proyecto se pudieron establecer las siguientes conclusiones:

- Es posible usar herramientas de generación de código como Ingravity para realizar las tareas repetitivas de construcción de componentes base para las aplicaciones
- La herramienta de generación de código Ingravity no genera la totalidad del código de la aplicación. Genera alrededor del 40% del código final, por lo que se necesita también construir código de forma tradicional.
- Para poder construir el código de la aplicación es necesario conocer y comprender el funcionamiento de la arquitectura ASAP sobre la cual trabaja este generador.
- La arquitectura de software define un esquema de organización del código en donde los diferentes tipos de componentes deben construirse y colocarse de una forma específica.
- Aunque al principio puede ser un poco confuso o complejo para el desarrollador, una vez se acostumbra a la forma de trabajo, facilita el desarrollo y mantenimiento de las aplicaciones.
- El cambio en la versión del generador y el framework redundan en reprocesos y nuevas capacitaciones que pueden ser costosas para la compañía.

En cuanto al trabajo desarrollado en la empresa Ingacon S.A., debido a sus propias restricciones involucró el desarrollo de tareas adicionales al desarrollo de software. Tareas que involucraron soporte a usuario con respecto al software C.O.D.E(Control y Organización para la Dirección Estratégica) y al hardware de la empresa, manejo de la red y resolución de problemas de diferentes tipos.

Sobre este trabajo, se pudieron establecer las siguientes conclusiones:

- Algunas empresas con limitaciones de recursos y personal, optan por hacer rotación del personal, asignándole diferentes tipos de tareas del grupo de ingenieros que compone el departamento de sistemas
- El soporte a los usuarios requiere de habilidades y conocimientos que no son solo técnicos, sino de comunicación y relación con la gente.

- El soporte requiere, en muchas ocasiones, de trabajar con usuarios de diferentes niveles de conocimiento y de alguna forma diferentes culturas. Situación que demanda de un mayor nivel de esfuerzo y compromiso.

Por otra parte, el desarrollo del proyecto resultó positivo también para Ingacon S.A, debido a que los módulos desarrollados durante la pasantía brindaron solución a las necesidades existentes y el constante uso por parte de los usuarios, de las aplicaciones, dio la credibilidad suficiente al proyecto C.O.D.E

12.RECOMENDACIONES

Siendo la internet y el desarrollo de aplicaciones web una constante hoy en día, es una buena alternativa para una empresa, tomar la decisión de hacer el desarrollo de sus aplicaciones empresariales usando tecnologías web. En el caso de Ingacon S.A favorece la creación de una aplicación de este tipo por el hecho de estar situada en distintas localidades del país, dado que esto permite el fácil acceso de los usuarios a la aplicación y la disponibilidad continua del aplicativo, visitándolo desde el lugar de trabajo o incluso desde casa.

Para el desarrollo de software se ha popularizado el utilizar programas gratuitos o de código fuente libre, que redundan en economía para la empresa. Es posible que empresas usen programas como el servidor web de Apache, el lenguaje de programación PHP y el sistema de bases de datos PostgreSQL, para el desarrollo de sus aplicaciones empresariales.

Se recomienda el uso de herramientas para el modelamiento de la base de datos y del software. Herramientas comerciales como Sybase Power Designer o gratuitas como DeSign, pueden servir para este propósito. Estas herramientas agilizan las tareas que realiza el desarrollador y son muy fáciles de aprender.

Se recomienda emplear un poco más de tiempo en el proceso de planeación para la creación de módulos, ya que así pueden obtenerse mejores resultados, asegurándose que todos los requerimientos están cubiertos y que los requerimientos implícitos también se tengan en cuenta. Adicionalmente, considerando que se está modificando un sistema de información que ya está funcionando, se puede obtener una visión más precisa de las relaciones o tablas necesarias a incluirse en la Base de Datos ya existente.

GLOSARIO

AJAX: (Asynchronous JavaScript and XML). Técnica para el desarrollo de aplicaciones web, en donde la aplicación realiza peticiones asincrónicas a un servidor web, de forma que la interfaz de usuario pueda actualizarse parcialmente sin refrescar toda la página. Esta técnica permite construir aplicaciones interactivas de mayor rendimiento, combinando el HTML con programas en Javascript, y datos en XML o JSON. Derivado de una técnica similar conocida como “Remote Scripting” liberada por Microsoft en 1998 y en el proyecto opensource JSRS en 2000, se ha popularizado desde la aparición del término en 2005.[1]

APLICACIÓN WEB: Aplicación de software que se codifica en un lenguaje soportado por los navegadores web (HTML, JavaScript, Java, etc.) y que se accede utilizando servidor web a través de Internet o intranet. Una estrategia utilizada por las empresas para las proveer acceso al software a través de internet desde cualquier localización y no solo dentro de la misma empresa.

ASAP: Arquitectura de software aplicada para el desarrollo de aplicaciones web usando bases de datos y el lenguaje PHP. Arquitectura de tres capas que divide el software en tres grandes bloques, cada uno de éstos con una funcionalidad definida: presentación visual, lógica de dominio y acceso a datos, adaptándose a las necesidades de cambio rápido en la aplicación, dando solución a los problemas de diseño de software. Esta arquitectura cuenta con el generador de código Ingravity. Resultado de un proyecto de investigación del Laboratorio de Investigaciones para el Desarrollo de la Ingeniería de Software LIDIS de la Universidad de San Buenaventura Cali, en colaboración con empresas de ParqueSoft Cali.

CSS: (Cascading Style Sheets) Hojas de estilo en cascada, mecanismo de definición de estilo que permite a los desarrolladores establecer la apariencia visual de una página web, incluyendo sus documentos, sus textos, imágenes y la disposición geométrica de sus componentes.

GESTION DE INVENTARIOS: Proceso de negocio relacionado con la validación, administración y control de los productos en inventario de una Empresa. Proceso de gran importancia, debido a que la gestión apropiada los niveles de inventarios es la base de varias estrategias de negocio, incluyendo el marco de la producción ajustada (Lean Production). Un sistema eficiente de administración de inventarios, puede contribuir a producir economías de escala, liberar capital de trabajo, reducir los costos de los productos, mejorar el nivel de respuesta al cliente y la competitividad de la empresa. [5]

HTTP: (Hypertext Transfer Protocol) Protocolo de Transferencia de Hipertexto. Protocolo de red, ampliamente usado en Internet, empleado en los servicios de World Wide Web (WWW) para la transferencia de documentos HTML e imágenes entre navegadores y servidores web. Es un protocolo de nivel de aplicación, definido en el RFC-2616 del Internet Engineering Task Force (IETF)¹, orientado hacia la comunicación entre sistemas de información distribuidos, colaborativos.

INVENTARIO PERIÓDICO: Modelo de Inventario generalmente utilizado por empresas medianas y pequeñas. La base del sistema de inventario periódico es el conteo físico de las mercancías disponibles al final del periodo. Consiste en la realización de un inventario de forma que se recuenten físicamente las existencias que sean propiedad de la empresa, independientemente de su localización.

MODELO VISTA CONTROLADOR: Patrón de diseño empleado para establecer la arquitectura de software para el desarrollo de la interfaz de usuario de una aplicación. En el patrón de modelo-vista-controlador, se definen tres diferentes tipos de componentes: Componentes del Modelo, de la Vista y los controladores.

- **Los componentes del Modelo**, contienen la funcionalidad y el estado de la aplicación. En él se define la estructura de los datos relevantes al dominio y se definen las funciones que procesan esos datos.
- **Los componentes de la Vista**, proporcionan la interfaz del modelo y los mecanismos de interacción con el usuario. Normalmente la vista incluye componentes que generan la presentación visualiza el cliente. En el caso de las aplicaciones web, puede incluir archivos HTML, applets y código JavaScript, entre otros.
- **Los controladores**, establecen la conexión entre los elementos del interfaz y los datos que estos representan. El controlador procesa las peticiones de los usuarios, implementa el flujo de la aplicación y de acuerdo al estado actual del modelo, ejecuta alguno de los componentes de la vista.

PLUGIN: Producto o componente de software adicional que permite agregar funcionalidades nuevas a una aplicación, Por ejemplo, en los navegadores web, un plugin puede permitir funcionalidades adicionales como la reproducción de sonidos, video o aplicaciones.

PHP: Lenguaje de programación de código fuente abierto, usado primordialmente para el desarrollo de aplicaciones web [7]. Fue uno de los primeros lenguajes que permitían construir programas embebidos en el código HTML de las páginas web. Creado a principios de 1995 por Rasmus Lerdorf, fue rediseñado completamente en 1998 por Zeev Suraski y Andi Gutmans en la versión PHP 3. A mediados de

¹Network Working Group. Fielding et al. RFC 2616: Hypertext Transfer Protocol – Http/1.1.[en línea]: HTTP. Estados Unidos: The Internet Society. 1999.[Consultado 26 Noviembre 2009]. Disponible en <http://www.ietf.org/rfc/rfc2616.txt>

2004 fue rediseñado nuevamente para incorporar características de orientación a objetos en PHP 5.

POSTGRES o POSTGRESQL: Sistema de base de datos objeto-relacional de código fuente abierto². Basado en el sistema original Ingres de la Universidad de California en Berkeley, Michael Stonebracker inició su desarrollo en 1985, liberando varias versiones a nivel científico durante varios años, hasta su publicación en código fuente abierto en 1996. En la actualidad cuenta con una comunidad de usuarios bastante amplia y existen varias empresas como Pervasive, Greenplum y EnterpriseDB que proveen productos comerciales basados en PostgreSQL.

PQR: Abreviatura usada para referirse a los sistemas de registro y seguimiento de Peticiones, Quejas y Reclamos. En el caso del proyecto, el sistema de PQR refiere a las actividades básicas del área de atención al cliente desarrolladas en las oficinas comerciales que responden a las necesidades de los clientes.

- **Peticiones**, son todas aquellas solicitudes que el cliente utiliza para que se le atienda acorde a sus necesidades con relación al servicio.
- **Quejas**, son el medio por el cual el usuario o suscriptor pone de manifiesto su inconformidad con la actuación de determinado (s) empleado (s), o su inconformidad con la forma y condiciones en que se ha prestado el servicio.
- **Reclamos**, son oposiciones que se formulan a la organización. Exigencia de los derechos del cliente y/o usuario, relacionada con la prestación de los servicios que se ofrecen.

PROGRAMA DE SALUD OCUPACIONAL: Programa que deben establecer las empresas para realizar actividades de prevención y atención en salud de sus trabajadores. Los programas de Salud Ocupacional son obligatorios en Colombia desde la Resolución 1016 de 1989, tomando como base varios artículos y reglamentaciones establecidos desde 1979.

Un Programa de Salud Ocupacional, consiste en la planeación y ejecución de actividades de medicina, seguridad e higiene industrial, capacitaciones, que tienen como objetivo mantener y mejorar la salud de los trabajadores en las empresas. Actividades que deben ser monitoreadas por un Comité Paritario de Salud Ocupacional, del cual forman parte algunos empleados elegidos por el conjunto de trabajadores.

El Programa de Salud Ocupacional, es desarrollado por la empresa con la participación de los trabajadores. Su ejecución debe ser permanente, y debe ser

² PostgreSQL Global Development Group. PostgreSQL. [en línea]: About. Berkeley: *PostgreSQL Global Development Group*) 2010 [Consultado 26 Noviembre 2009] Disponible en Internet: <http://www.postgresql.org/>

concebido como un proceso de mejoramiento continuo de las condiciones de trabajo.

XAJAX: Librería de clases de código abierto en PHP para agregar funcionalidades de AJAX en aplicaciones Web con tecnología AJAX³. La librería facilita la

construcción de aplicaciones usando la técnica de AJAX, registrando clases y funciones en el servidor usando PHP e invocándolas remotamente desde el navegador web usando Javascript. El código en Javascript puede obtener los resultados y actualizar la interfaz de usuario sin recargar toda la página.

³WHITE,Jared. XAJAX [en línea]: Tutorial: learn xajax in 10 minutes. Estados Unidos: Xajax Project, 2010 [Consultado 26 de Noviembre de 2009]. Disponible en Internet: <http://xajaxproject.org/>

BIBLIOGRAFÍA

- [1] ASLESON. Ryan. Pro AJAX and Java Frameworks, Editorial Apress, 2006 Capítulo 1.
- [2] COLCIENCIAS. Peticiones, Quejas y Reclamos [en línea]: PQR. Bogotá, Colombia: Departamento Administrativo de Ciencia, Tecnología e Innovación, 2010 [Consultado 12 de abril del 2009]. Disponible en Internet: http://www.colciencias.gov.co/portalcol/kernel/mod_pgrs/plantillestatica.jsp
- [3] Consejo Colombiano de Seguridad. Software [en línea]: PASO. Bogotá, Colombia: Consejo Colombiano de Seguridad, 2004 [Consultado 12 de abril 2009]. Disponible en Internet: <http://www.laseguridad.ws/consejo/consejo/home.htm>
- [4] Dirección General de Sanidad Militar. Peticiones, Quejas y Reclamos [En línea] Crear Solicitud. Bogotá, Colombia: Sanidad Fuerzas Militares, 2007 [Consulta 01 de junio de 2009] Disponible en Internet: <http://www.sanidadfuerzasmilitares.mil.co/?idcategoria=6>
- [5] EHULETCHE, Ana Belén. Administración de inventarios: eficacia y rentabilidad [en línea]: Gestión de Inventarios. México: Revista conciencia y satisfacción en un nuevo espacio de negocio, 2009 [Consultado 24 de Noviembre del 2009]. Disponible en Internet: <http://www.logistica.enfasis.com/notas/14710-administracion-inventarios-eficacia-y-rentabilidad>
- [6] FLOWER, Martin. Patterns of Enterprise Application Architecture. The Addison-Wesley Signature Series, 2002. 560 p.
- [7] GIL RUBIO. F. Javier, VILLAVERDE. Alonso, TEJEDOR. Jorge A, YAGUE PANADERO. Agustín, Creación de Sitios Web con PHP5, Editorial McGraw Hill, 2006 Capítulo 1.
- [8] HASSAN, Yusef. Patrones de Diseño de Interacción [en línea]: Colecciones y Lenguajes de Patrones. Argentina: SIDAR, 2005 [Consultado el 15 de mayo 2009]. Disponible en Internet: <http://www.bitacoras.sidar.org/g4/index.php?2005/09/22/10-patrones-de-diseno-de-interaccion-iii-colecciones-y-lenguajes-de-patrones>
- [9] LIDIS. Arquitectura [en línea]: Arquitectura LIDIS. Bogotá, Colombia: PalmeraTech Wikispaces, 2009 [Consultado 12 de Noviembre 2009]. Disponible en Internet: <http://palmeratech.wikispaces.com/ARQUITECTURA>

[10] REINOSO, Carlos. Introducción a la Arquitectura de Software [en línea]: Arquitectura de Software. Buenos Aires: WillyDev.net , 2008 [Consultado 16 de Marzo de 2010]. Disponible en Internet: <http://www.willydev.net/descargas/prev/IntroArq.pdf>

[11] VALDES, Damián. ¿Qué son las bases de datos? [en línea]: Definición de base de datos. México: Maestros del Web, 2007 [Consultado 15 de mayo 2009]. Disponible en Internet: <http://www.maestrosdelweb.com/principiantes/%C2%BFque-son-las-bases-de-datos/>

[12] VELARDE, Peter. Soluciones Web [en línea]: Solución para captar nuevos clientes vía web. Lima, Perú: Innova Soluciones Informáticas, 2007 [Consultado 12 de abril de 2009] Disponible en Internet: <http://www.mundoinnova.net/solucionesweb.htm>

ANEXOS

Anexo 1. Modelo Entidad Relación

Anexo 1.1 MER Inventario Periódico

Anexo 1.2 MER PQR

Anexo 1.3. MER S.O

Anexo 2. Modelo Relacional de Datos

Anexo 2.1. MRD Inventario Periódico

Anexo 2.2. MRD PQR

Anexo 2.3. MRD Programación ACT S.O

Anexo 3. Descripción o Guiones de Casos de Uso

Anexo 3.1 Guion Caso de Uso Inventario Periódico

Anexo 3.2 Guion Caso de Uso Programación Actividad S.O

Anexo 3.3 Guion Caso de Uso PQR

Anexo 4. Diagrama Relacional de Casos de Uso

Anexo 4.1 Diagrama Relacional C.U Inventario Periódico

Anexo 4.2 Diagrama Relacional C.U PQR

Anexo 4.2 Diagrama Relacional C.U Programación Actividad S.O

Anexo 5. Preliminares

Anexo 5.1 Preliminares Inventario Periódico

Anexo 5.2 Preliminares PQR

Anexo 5.3 Preliminares Programación Actividad Salud Ocupacional

Anexo 6. Cartas

Anexo 6.1 Carta Director del Proyecto

Anexo 6.2 Carta Aceptación del Proyecto Ingacon S.A

Anexo 6.3 Formato de Evaluación CRC Elaborado por Ingacon S.A